

IN THE CLAIMS:

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A computer-implemented method for programmatically generating a graphical program, the method comprising:

displaying a graphical user interface (GUI) on a display;

receiving user input to the GUI specifying desired functionality of the graphical program; and

[[programmatically]] automatically generating the graphical program in response to the user input specifying the functionality of the graphical program, wherein the graphical program implements the specified functionality;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program;
and

wherein said [[programmatically]] automatically generating the graphical program includes generating the block diagram portion ~~and the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

2. (Original) The method of claim 1,

wherein the GUI comprises information useable in guiding a user in creation of a program.

3. (Original) The method of claim 1,

wherein the GUI comprises one or more GUI input panels;

wherein the user input to the GUI comprises user input to each of the one or more GUI input panels.

4. (Original) The method of claim 3,

wherein said displaying the GUI and said receiving user input to the GUI comprise:

displaying a first GUI input panel on the display, wherein the first GUI input panel includes one or more first fields adapted to receive user input specifying first functionality of the graphical program;

receiving first user input specifying first functionality of the graphical program;

displaying a second GUI input panel on the display, wherein the second GUI input panel includes one or more second fields adapted to receive user input specifying second functionality of the graphical program;

receiving second user input specifying second functionality of the graphical program.

5. (Original) The method of claim 4,

wherein the second GUI input panel is one of a plurality of possible second GUI input panels, wherein the second GUI input panel is displayed based on the first user input.

6. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical program comprises [[programmatically]] automatically generating a portion of a graphical program.

7. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical program creates the graphical program without any user input specifying the new graphical program during said creating.

8. (Cancelled).

9. (Cancelled)

10. (Currently Amended) The method of claim 1, wherein said [[programmatically]] automatically generating the graphical program comprises:

automatically creating a plurality of graphical program objects in the graphical program; and

automatically interconnecting the plurality of graphical program objects in the graphical program;

wherein the interconnected plurality of graphical program objects comprise at least a portion of the graphical program.

11. (Currently Amended) The method of claim 1, wherein said [[programmatically]] automatically generating the graphical program comprises:

automatically creating one or more user interface objects in the graphical program, wherein the one or more user interface objects perform one or more of providing input to or displaying output from the graphical program.

12. (Currently Amended) The method of claim 1,
wherein the user input received specifies an instrumentation function;
wherein the [[programmatically]] automatically generated graphical program implements the specified instrumentation function.

13. (Original) The method of claim 12,

wherein the instrumentation function comprises one or more of:

a test and measurement function; or

an industrial automation function.

14. (Currently Amended) The method of claim 1,

wherein said [[programmatically]] automatically generating the graphical program comprises calling an application programming interface (API) enabling the programmatic generation of a graphical program.

15. (Currently Amended) The method of claim 1,
wherein said ~~[[programmatically]]~~ automatically generating the graphical program comprises ~~[[programmatically]]~~ automatically requesting a server program to generate the graphical program.

16. (Currently Amended) A computer-implemented method for ~~[[programmatically]]~~ automatically generating a graphical program, the method comprising:

displaying a plurality of GUI input panels on a display, wherein the GUI input panels comprise information useable in guiding a user in creation of a program;

receiving user input to the plurality of GUI input panels, wherein the user input specifies desired functionality of the graphical program; and

~~[[programmatically]]~~ automatically generating the graphical program in response to the user input specifying the functionality of the graphical program, wherein the graphical program implements the specified functionality;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program;
and

wherein said ~~[[programmatically]]~~ automatically generating the graphical program includes generating the block diagram portion ~~and the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

17. (Currently Amended) A computer-implemented method for ~~[[programmatically]]~~ automatically generating a graphical program, the method comprising:

displaying a graphical user interface (GUI) on a display;

receiving user input to the GUI specifying desired functionality of the graphical program;

executing a graphical program generation (GPG) program;

the GPG program receiving the user input, wherein the user input specifies the desired functionality of the new graphical program; and

the GPG program ~~[[programmatically]]~~ automatically generating the graphical program in response to the user input specifying the functionality of the graphical program, wherein the graphical program implements the specified functionality;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program;
and

wherein said ~~[[programmatically]]~~ automatically generating the graphical program includes generating the block diagram portion ~~and the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

18. (Currently Amended) A computer-implemented method for ~~[[programmatically]]~~ automatically generating a graphical program, the method comprising:

displaying a graphical user interface (GUI) on a display;

receiving user input to the GUI indicating desired program operation of the graphical program;

executing a graphical program generation (GPG) program;

the GPG program receiving the user input, wherein the user input indicates desired operation of the graphical program; and

the GPG program ~~[[programmatically]]~~ automatically generating the graphical program in response to the user input indicating the desired operation of the graphical program, wherein the graphical program implements the desired operation;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program;
and

wherein said ~~[[programmatically]]~~ automatically generating the graphical program includes generating the block diagram portion ~~and the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

19. (Original) The method of claim 18,

wherein the GPG program comprises a graphical programming development environment application.

20. (Original) The method of claim 18,

wherein the GPG program is operable to generate a plurality of graphical programs, depending on the received user input.

21. (Currently Amended) A computer-implemented method for ~~[[programmatically]]~~ automatically generating a graphical program, the method comprising:

displaying one or more input panels on a display;

receiving user input to the one or more input panels; and

~~[[programmatically]]~~ automatically generating graphical source code for the graphical program, based on the received user input;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program;
and

wherein said ~~[[programmatically]]~~ automatically generating graphical source code for the graphical program includes generating graphical source code for the block

diagram portion ~~and for the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

22. (Currently Amended) The method of claim 21,
wherein the one or more input panels comprise a graphical user interface (GUI) useable in guiding a user in specifying program functionality;
wherein the received user input specifies desired functionality of the graphical program;
wherein the ~~[[programmatically]]~~ automatically generated graphical source code implements the specified desired functionality.

23. (Currently Amended) A computer-implemented method for ~~[[programmatically]]~~ automatically generating a graphical program, the method comprising:

displaying a node in the graphical program in response to user input;
displaying a graphical user interface (GUI) for configuring functionality for the node in response to user input;

receiving user input via the GUI indicating desired functionality for the node; and
~~[[programmatically]]~~ automatically including graphical source code associated with the node in the graphical program, wherein the ~~[[programmatically]]~~ automatically included graphical source code implements the desired functionality, and wherein the graphical source code comprises a plurality of interconnected nodes that visually represent the desired functionality;

wherein said ~~[[programmatically]]~~ automatically including graphical source code associated with the node in the graphical program comprises ~~[[programmatically]]~~ automatically including the graphical source code as a sub-program of the graphical program without direct user input specifying the plurality of nodes or connections between the nodes, wherein the node represents the sub-program.

24. (Cancelled)

25. (Cancelled)

26. (Currently Amended) A method for configuring a node in a graphical program, the method comprising:

displaying the node in the graphical program;

displaying a graphical user interface (GUI) associated with the node, wherein the GUI comprises information useable in guiding a user in specifying desired functionality for the node;

receiving user input to the GUI specifying desired functionality for the node; and

[[programmatically]] automatically generating graphical source code associated with the node to implement the specified functionality, wherein the graphical source code comprises a plurality of interconnected nodes that visually represent the desired functionality;

wherein said [[programmatically]] automatically generating graphical source code associated with the node comprises [[programmatically]] automatically generating the graphical source code as a sub-program of the graphical program without direct user input specifying the plurality of nodes or connections between the nodes, wherein the node represents the sub-program.

27. (Cancelled)

28. (Cancelled)

29. (Currently Amended) The method of claim 26,

wherein no graphical source code is associated with the node until after said [[programmatically]] automatically generating graphical source code associated with the node.

30. (Currently Amended) The method of claim 26,

wherein default graphical source code is associated with the node;

wherein said [[programmatically]] automatically generating graphical source code associated with the node comprises replacing the default graphical source code with the [[programmatically]] automatically generated graphical source code.

31. (Currently Amended) The method of claim 26,

wherein no functionality is defined for the node until after said [[programmatically]] automatically generating graphical source code associated with the node.

32. (Currently Amended) The method of claim 26,

wherein no program instructions to be executed during execution of the graphical program are associated with the node until after said [[programmatically]] automatically generating graphical source code associated with the node.

33. (Currently Amended) The method of claim 26, further comprising:

receiving user input requesting to change functionality of the node, after said [[programmatically]] automatically generating the graphical source code;

re-displaying the GUI in response to the user input requesting to change functionality of the node;

receiving user input to the GUI specifying new functionality for the node;

[[programmatically]] automatically replacing the previously generated graphical source code with new graphical source code to implement the new functionality for the node.

34. (Currently Amended) A memory medium for [[programmatically]] automatically generating a graphical program, the memory medium comprising program instructions executable to:

display a graphical user interface (GUI) on a display;

receive user input to the GUI specifying desired functionality of the graphical program; and

[[programmatically]] automatically generate the graphical program in response to the user input specifying the functionality of the graphical program, wherein the graphical program implements the specified functionality;

wherein the graphical program comprises a block diagram portion comprising a plurality of interconnected nodes, and a graphical user interface portion, wherein the plurality of interconnected nodes visually indicate functionality of the graphical program; and

wherein said [[programmatically]] automatically generating the graphical program includes generating the block diagram portion ~~and the graphical user interface portion~~ without direct user input specifying the plurality of nodes or connections between the nodes.

35. (Original) The memory medium of claim 34,
wherein the GUI comprises information useable in guiding a user in creation of a program.

36. (Original) The memory medium of claim 34,
wherein the GUI comprises one or more GUI input panels;
wherein the user input to the GUI comprises user input to each of the one or more GUI input panels.

37. (Original) The memory medium of claim 36, wherein the program instructions executable to display the GUI and the program instructions executable to receive user input to the GUI comprise program instructions executable to:

display a first GUI input panel on the display, wherein the first GUI input panel includes one or more first fields adapted to receive user input specifying first functionality of the graphical program;

receive first user input specifying first functionality of the graphical program;

display a second GUI input panel on the display, wherein the second GUI input panel includes one or more second fields adapted to receive user input specifying second functionality of the graphical program;

receive second user input specifying second functionality of the graphical program.

38. (Original) The memory medium of claim 37,
wherein the second GUI input panel is one of a plurality of possible second GUI input panels, wherein the second GUI input panel is displayed based on the first user input.

39. (Cancelled).

40. (Currently Amended) The memory medium of claim 34,
wherein said [[programmatically]] automatically generating the graphical program creates the graphical program without any user input specifying the new graphical program during said creating.

41. (Cancelled).

42. (Cancelled)

43. (Currently Amended) The memory medium of claim 34, wherein the program instructions executable to [[programmatically]] automatically generate the graphical program comprise program instructions executable to:

automatically create a plurality of graphical program objects in the graphical program; and

automatically interconnect the plurality of graphical program objects in the graphical program;

wherein the interconnected plurality of graphical program objects comprise at least a portion of the graphical program.

44. (Currently Amended) The memory medium of claim 34,
wherein the user input received specifies an instrumentation function;
wherein the [[programmatically]] automatically generated graphical program
implements the specified instrumentation function.

45. (Original) The memory medium of claim 44,
wherein the instrumentation function comprises one or more of:
a test and measurement function; or
an industrial automation function.

46. (Currently Amended) A memory medium for configuring a node in a
graphical program, the memory medium comprising program instructions executable to:
display the node in the graphical program;
display a graphical user interface (GUI) associated with the node, wherein the
GUI comprises information useable in guiding a user in specifying desired functionality
for the node;

receive user input to the GUI specifying desired functionality for the node; and
[[programmatically]] automatically generate graphical source code associated
with the node to implement the specified functionality, and wherein the graphical source
code comprises a plurality of interconnected nodes that visually represent the desired
functionality;

wherein said [[programmatically]] automatically generating graphical source code
associated with the node comprises [[programmatically]] automatically generating the
graphical source code as a sub-program of the graphical program without direct user
input specifying the plurality of nodes or connections between the nodes, wherein the
node represents the sub-program.

47-48. (Cancelled)

49. (Currently Amended) The memory medium of claim 46,
wherein no graphical source code is associated with the node until after said
[[programmatically]] automatically generating graphical source code associated with the
node.

50. (Currently Amended) The memory medium of claim 46,
wherein default graphical source code is associated with the node;
wherein said [[programmatically]] automatically generating graphical source code
associated with the node comprises replacing the default graphical source code with the
[[programmatically]] automatically generated graphical source code.

51. (Currently Amended) The memory medium of claim 46, further comprising
program instructions executable to:

receive user input requesting to change functionality of the node, after said
[[programmatically]] automatically generating the graphical source code;

re-display the GUI in response to the user input requesting to change functionality
of the node;

receive user input to the GUI specifying new functionality for the node;

[[programmatically]] automatically replace the previously generated graphical
source code with new graphical source code to implement the new functionality for the
node.

52. (Previously Presented) The method of claim 1, wherein the graphical user
interface portion comprises a front panel, wherein the front panel comprises one or more
controls and/or one or more indicators.

53. (Previously Presented) The memory medium of claim 34, wherein the
graphical user interface portion comprises a front panel, wherein the front panel
comprises one or more controls and/or one or more indicators.